

3D Menagerie: Modeling the 3D shape and pose of animals

Silvia Zuffi¹ Angjoo Kanazawa² David Jacobs² Michael J. Black³

¹IMATI-CNR, Milan, Italy, ²University of Maryland College Park

³Max Planck Institute for Intelligent Systems, Tübingen, Germany

silvia@mi.imati.cnr.it, {kanazawa, djacobs}@umiacs.umd.edu, black@tuebingen.mpg.de

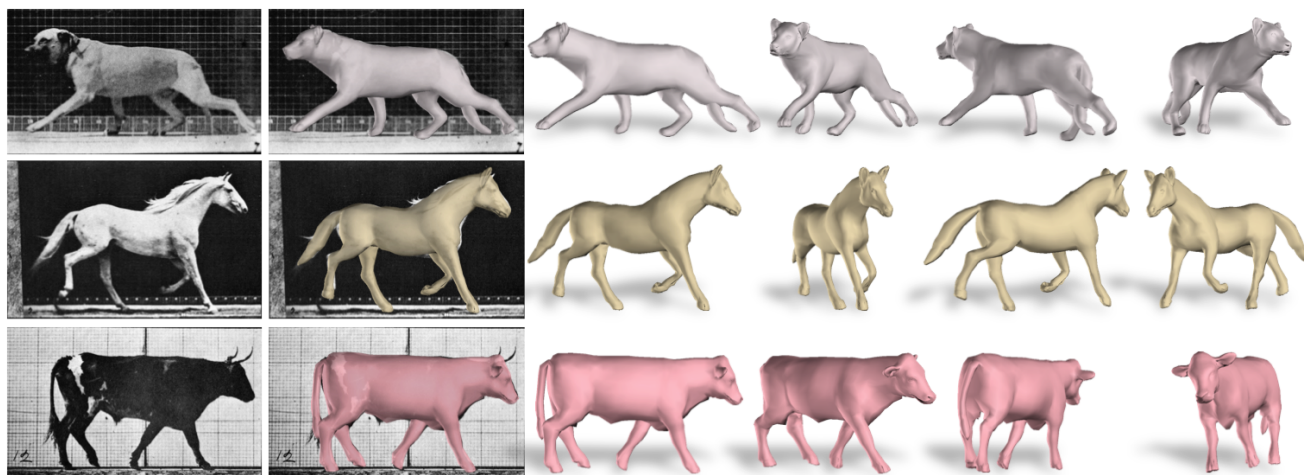


Figure 1: **Animals from Images.** We learn an articulated, 3D, statistical shape model of animals using very little training data. We fit the shape and pose of the model to 2D image cues showing how it generalizes to previously unseen shapes (Photos Eadweard Muybridge).

Abstract

There has been significant prior work on learning realistic, articulated, 3D statistical shape models of the human body. In contrast, there are few such models for animals, despite their many applications. The main challenge is that animals are much less cooperative subjects than humans. The best human body models are learned from thousands of 3D scans of people in specific poses, which is infeasible with live animals. Consequently, here we extend a state-of-the-art articulated 3D human body model to animals and learn it from a limited set of 3D scans of toy figurines in arbitrary poses. We employ a novel part-based shape model to compute an initial registration to the scans. We then normalize their pose, learn a statistical shape model, and refine the alignments and the model together. In this way, we accurately align animal scans from different quadruped families with very different shapes and poses. With the alignment to a common template we learn a shape space representing animals including lions, cats, dogs, horses, cows and hippos. Animal shapes can be sampled from the model, posed, animated, and fitted to data. In particular, we demonstrate

the generalization of the model by fitting it to images of real animals, and show that it captures realistic animal shapes, even for new species not seen in training. We make our model available for research, enabling the extension of methods for human shape and pose estimation to animals.

1. Introduction

The detection, tracking, and analysis of animals has many applications in biology, neuroscience, ecology, farming, and entertainment. Despite the wide applicability, the computer vision community has focused more heavily on modeling humans, estimating human pose, and analyzing human behavior. Can we take the best practices learned from the analysis of humans and apply these directly to animals? To address this, we take an approach for 3D human pose and shape modeling and extend it to modeling animals. We find that modeling animals presents novel challenges and describe how to overcome these.

Specifically we learn a generative model of the 3D pose and shape of animals and then fit this model to 2D image data as illustrated in Fig. 1. We focus on a subset of four-legged mammals that all have the same number of “parts” and model members of the families Felidae, Canidae, Equidae, Bovidae, and Hippopotamidae. Our goal is to build a statistical shape model like SMPL [20], which captures human body shape variation in a low-dimensional Euclidean subspace, models the articulated structure of the body, and can be fit to image data [6].

Animals, however, differ from humans in several important ways. First, the shape variation across species far exceeds the kinds of variations seen between humans. Even within the canine family, there is a huge variability in dog shapes as a result of selective breeding. Second, all these animals have tails, which are highly deformable and obviously not present in human shape models. Third, obtaining 3D data to train a model is much more challenging. SMPL and previous models like it (e.g. SCAPE [4]) rely on a large database of thousands of 3D scans of many people (capturing shape variation in the population) and a wide range of poses (capturing pose variation). Humans are particularly easy and cooperative subjects. It is impractical to bring a large number of wild animals into a lab environment for scanning and it would be difficult to take scanning equipment into the wild to capture animals shapes in nature.

Since scanning live animals is impractical we instead scan realistic toy animals to create a dataset of 41 scans of a range of quadrupeds as illustrated in Fig. 2. We show that a model learned from toys generalizes to real animals. The key to building a statistical 3D shape model is that all the 3D data must be in correspondence. This involves registering a common template mesh to every scan. This is a hard problem, which we approach in a series of steps.

Step 1, coarse registration: We begin with an artist-designed template mesh that is manually segmented and rigged for animation with a skeleton and blend weights (Fig. 4). Aligning a single template shape like this to a wide variety of animal shapes is challenging. Consequently, we introduce a new approach for part-based modeling and inference that extends the “stitched puppet” (SP) model [30]. We represent the body by a set of loosely connected parts where each part has an analytic shape space that lets it deform to match previously seen animals. Unlike SP, we use a global optimization of the part shapes and poses and minimize a richer energy function. This new *Global-Local Stitched Shape* model (GLoSS) provides a coarse registration between very different animal shapes.

Step 2, ARAP refinement: The GLoSS alignments are somewhat crude but provide a reasonable initialization for refinement. Consequently the GLoSS alignment is followed by a *model-free* step where the template mesh vertices deform towards the scan surface under a As-Rigid-As-

Possible (ARAP) constraint [27].

Step 3, Initial shape modeling: We use the articulated nature of the model and the initial alignment to the scan to “pose normalize” the scans. We define a common pose that has the legs and tail extended. Once all scans are in a common pose, we use principal component analysis (PCA) to construct a low-dimensional shape model of the vertices. This is analogous to the SMPL shape-blend shape space [20]. Using the articulated structure of the template and its blend weights, we now obtain a model where new shapes can be generated and reposed.

Step 4, Co-registration: Finally we refine the alignment of the template to the scans using co-registration [16], which regularizes the registration by penalizing deviations from the model fit to the scan. We then update the shape space and iterate.

The final *Skinned Multi-Animal Linear* model (SMAL) provides a shape space of animals trained from 41 scans. We also leverage this to model distributions conditioned on particular animal classes. Because quadrupeds have shape variations in common, the model generalizes to new animals not seen in training. This allows us to fit SMAL to 2D data using manually detected keypoints and segmentations. As shown in Fig. 1 and Fig. 10 our model can generate realistic animal shapes in a wide variety of poses.

In summary we describe a method to create realistic 3D models of animals and fit this model to 2D data. The problem is much harder than modeling humans and we develop new tools to extend previous methods to learn an animal model. This opens up new directions for research on animal shape and motion capture. To facilitate this, the animal model is made available for research purposes.

2. Related work

There is a long history on representing, classifying, and analyzing animal shapes in 2D [28]. Here we focus only on work in 3D. The idea of part-based 3D models of animals also has a long history. Similar in spirit to our GLoSS model, Marr and Nishihara [22] suggested that a wide range of animals shapes could be modeled by a small set of 3D shape primitives connected in a kinematic tree.

Animal Shape from 3D Scans. There is little work that systematically addresses the 3D scanning [2] and modeling of animals. The range of sizes and shapes, together with the difficulty of handling live animals and dealing with their movement, makes traditional scanning difficult. Previous 3D shape datasets like TOSCA [8] have a limited set of 3D animals that are artist-designed and with limited realism.

Animal Shape from images. Previous work on modeling animal shape starts from the assumption that obtaining 3D animal scans is impractical and focuses on using image data to extract 3D shape. Cashman and Fitzgibbon [9] take a template of a dolphin and learn a low-D model of its defor-



Figure 2: **Toys.** Example 3D scans of animal figurines used for training our model.

mations from hand clicked keypoints and manual segmentation. They optimize their model to minimize reprojection error to the keypoints and contour. They also show results for a pigeon and a somewhat cubist polar bear. The formulation is elegant but the approach suffers from an overly smooth shape representation; this is not so problematic for dolphins but for other animals it is. The key limitation, however, is that they do not model articulation.

Kanazawa et al. [17] deform a 3D animal template to match hand clicked points in a set of images. They learn separate deformable models for cats and horses using spatially varying stiffness values. Our model is stronger in that it captures articulation separately from shape variation. Further we model the shape variation across a wide range of animals to produce a statistical shape model.

Ntouskos et al. [23] take multiple views of different animals from the same class, manually segment the parts in each view, and then fit geometric primitives to segmented parts. They assemble these to form a crude 3D shape. Vicente and Agapito [29] extract a template from a reference image and then deforms it to fit a new image using keypoints and the silhouette. The results are of low resolution when applied to complex shapes.

Our work is complementary to these previous approaches that only use image data to learn 3D shapes. Future work should combine 3D scans with image data to obtain even richer models.

Animal Shape from Video. Ramanan et al. [24] model animals as a 2D kinematic chain of parts and learn the parts and their appearance from video. Bregler et al. [7] track features on a non-rigid object (e.g. a giraffe neck) and extract a 3D surface as well as its low-dimensional modes of deformation. Del Pero et al. [12] track and segment animals in video but do not address 3D shape reconstruction. Favreau et al. [13] focus on animating a 3D model of an animal given a 2D video sequence. Reinert et al. [25] take a video sequence of an animal and, using an interactive sketching/tracking approach, extract a textured 3D model of the animal. The 3D shape is obtained by fitting generalized cylinders to each sketched stroke over multiple frames.

None of these methods model the kinds of detail available in 3D scans, nor do they model the 3D articulated structure of the body. Most importantly none try to learn a 3D shape space spanning multiple animals.

Humans Shape from 3D Scans. Our approach is inspired by a long history of learning 3D shape models of humans. Blanz and Vetter [5] began this direction aligning 3D scans of faces and computing a low-D shape model. Faces have less shape variability and are less articulated than animals, simplifying mesh registration and modeling. Modeling articulated human body shape is significantly harder but several models have been proposed [3, 4, 15, 20, 11]. Chen et al. [10] model both humans and sharks, factoring deformations into pose and shape. The 3D shark model is learned from synthetic data and they do not model articulation.

We base our method on SMPL [20], which combines a low-D shape space with an articulated blend-skinned model. SMPL is learned from 3D scans of 4000 people in a common pose and another 1800 scans of 60 people in a wide variety of poses. In contrast, we have much less data and at the same time much more shape variability to represent. Despite this, we show that we can learn a useful animal model for computer vision applications. More importantly, this provides a path to making better models using more scans as well as image data. We also go beyond SMPL to add a non-rigid tail and more parts than are present in the human.

3. Dataset

We created a dataset of 3D animals by scanning toy figurines (Fig. 2) using an Artec hand-held 3D scanner. We also tried scanning taxidermy animals in a museum but found, surprisingly, that the shapes of the toys looked more realistic. We collected a total of 41 scans from several species: 1 cat, 5 cheetah, 8 lions, 7 tigers, 2 dogs, 1 fox, 1 wolf, 1 hyena, 1 deer, 1 horse, 6 zebras, 4 cows, 3 hippos. We estimated a scaling factor so animals from different manufacturers were comparable in size. Like previous 3D human datasets [26], and methods that create animals from images [9, 17], we collected a set of 36 hand-clicked

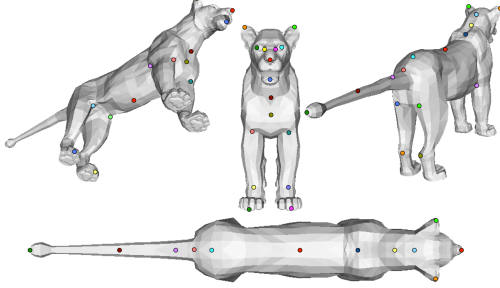


Figure 3: Location of 36 keypoints on the template mesh.

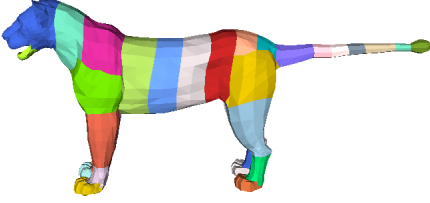


Figure 4: **Template mesh.** It is segmented into 33 parts, and here posed in the ‘‘T pose’’.

keypoints that we use to aid mesh registration. These are shown on the template in Fig. 3.

4. Global/Local Stitched Shape Model

The Global/Local Stitched Shape model (GLoSS) is a 3D articulated model where body shape deformations are locally defined for each part and the parts are assembled together by minimizing a stitching cost at the part interfaces. The model is inspired by the SP model [30], but has significant differences from it. In contrast to SP, the shape deformations of each part are analytic, rather than learned. This makes it more approximate but, importantly, allows us to apply it to novel animal shapes, without requiring *a priori* training data. Second, GLoSS is a globally differentiable model that can be fit to data with gradient-based techniques.

To define a GLoSS model we need the following: a 3D template mesh of an animal with the desired polygon count, its segmentation into parts, skinning weights, and an animation sequence. To define the mesh topology, we use a 3D mesh of a lioness from the Turbosquid website. The mesh is rigged and skinning weights are defined. We manually segment the mesh into $N = 33$ parts (Fig. 4) and make it symmetric along its main axis.

We now summarize the GLoSS parametrization. Let i be a part index, $i \in (1 \cdots N)$. The model variables are: part location $\mathbf{l}_i \in \mathbb{R}^{3 \times 1}$, part 3D rotation $\mathbf{r}_i \in \mathbb{R}^{3 \times 1}$, expressed as a Rodrigues vector, intrinsic shape variables $\mathbf{s}_i \in \mathbb{R}^{n_s \times 1}$ and pose deformation variables $\mathbf{d}_i \in \mathbb{R}^{n_d \times 1}$. The vector of vertex coordinates, $\hat{\mathbf{p}}_i \in \mathbb{R}^{3 \times n_i}$, for part i in a global reference frame is computed as:

$$\hat{\mathbf{p}}_i(\mathbf{l}_i, \mathbf{r}_i, \mathbf{s}_i, \mathbf{d}_i) = R(\mathbf{r}_i)\mathbf{p}_i + \mathbf{l}_i, \quad (1)$$

where n_i is the number of vertices in the part, and $R \in SO(3)$ is the rotation matrix obtained from \mathbf{r}_i . The $\mathbf{p}_i \in \mathbb{R}^{3 \times n_i}$ are points in local frame, computed as:

$$\mathbf{p}_i = \mathbf{t}_i + \mathbf{m}_{p,i} + B_{s,i}\mathbf{s}_i + B_{p,i}\mathbf{d}_i. \quad (2)$$

where $\mathbf{t}_i \in \mathbb{R}^{3n_i \times 1}$ is the part template, $\mathbf{m}_{p,i} \in \mathbb{R}^{3n_i \times 1}$ is the vector of average pose displacements; $B_{s,i} \in \mathbb{R}^{3n_i \times n_s}$ is a matrix with columns representing a basis of intrinsic shape displacements; and $B_{p,i} \in \mathbb{R}^{3n_i \times n_d}$ is the matrix of pose dependent deformations. These deformation matrices are defined below.

Pose deformation space. We compute the part-based pose deformation space from examples. For this we use an animation of the lioness template using linear blend skinning (LBS). Each frame of the animation is a pose deformation sample. We perform PCA on the vertices of each part in a local coordinate frame, obtaining a vector of average pose deformations $\mathbf{m}_{p,i}$ and the basis matrix $B_{p,i}$.

Shape deformation space. We define a synthetic shape space for each body part. This space includes 7 deformations of the part template, namely scale, scale along x , scale along y , scale along z , stretch based on x , stretch based on y and stretch based on z . This defines a simple analytic deformation for each part. The deformations are defined as follows:

$$\begin{aligned} \mathbf{b}_{scale_i} &= K_s \mathbf{t}_i \\ \mathbf{b}_{scale_{x,i}}(x) &= K_s \mathbf{t}_i(x); \mathbf{b}_{scale_{x,i}}(y) = \mathbf{b}_{scale_{x,i}}(z) = 0 \\ \mathbf{b}_{stretch_{x,i}}(x) &= \mathbf{t}_i(x) \\ \mathbf{b}_{stretch_{x,i}}(y) &= \mathbf{t}_i(y) K_s |\mathbf{t}_i(x)| \\ \mathbf{b}_{stretch_{x,i}}(z) &= \mathbf{t}_i(z) K_s |\mathbf{t}_i(x)|, \end{aligned} \quad (3)$$

and in a similar way for y and z . We assume each variable \mathbf{s}_i is Gaussian distributed with zero mean and variance that we set arbitrarily. For non uniform scaling we assign higher variance to the scaling along the ‘‘bone’’ of the part.

5. Initial Registration

The initial registration of the template to the scans is performed in two steps. First, we optimize a GLoSS model with a gradient-based method. This brings the model close to the scan. Then, we perform a model-free alignment of the mesh vertices to the scan using As-Rigid-As-Possible (ARAP) regularization [27] to capture the fine details.

GLoSS-based registration. To fit GLoSS to a scan, we minimize

$$E(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) = E_m(\mathbf{d}, \mathbf{s}) + E_{stitch}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) + E_{curv}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) + E_{data}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) + E_{pose}(\mathbf{r}), \quad (4)$$

where:

$$E_m(\mathbf{d}, \mathbf{s}) = k_{sm} E_{sm}(\mathbf{s}) + k_d \sum_{i=1}^N E_d(\mathbf{d}_i) + k_s \sum_{i=1}^N E_s(\mathbf{s}_i)$$

is a model term, where E_d and E_s express a prior over the values of the pose and shape deformations. This is defined as the squared Mahalanobis distance to Gaussian distributions with zero mean and variance given by the PCA eigenvalues for the pose deformation space and the variance defined for the synthetic model. The term E_{sm} represents the constraint that symmetric parts should have similar shape deformation values \mathbf{s} . We impose similarity between left and right leg parts and feet, back and front feet and sections of the torso. This last constraint favors sections of the torso to have similar length.

The stitching term E_{stitch} is the sum of squared distances of the corresponding points at the interfaces between parts (cf. [30]). $E_{stitch}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) =$

$$k_{st} \sum_{(i,j) \in \mathcal{C}} \|\hat{\mathbf{p}}_{I_{ij}}(\mathbf{l}_i, \mathbf{r}_i, \mathbf{s}_i, \mathbf{d}_i) - \hat{\mathbf{p}}_{I_{ji}}(\mathbf{l}_j, \mathbf{r}_j, \mathbf{s}_j, \mathbf{d}_j)\|^2, \quad (5)$$

where \mathcal{C} is the set of part connections, $\hat{\mathbf{p}}_{I_{ij}}$ is a vector of interface points between part i and part j , on part i ; $\hat{\mathbf{p}}_{I_{ji}}$ is a vector of interface points between part i and part j , on part j . Minimizing this term favors the parts to be connected.

The data term is defined as $E_{data}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) =$

$$E_{kp}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) + E_{m2s}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) + E_{s2m}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}), \quad (6)$$

where E_{m2s} and E_{s2m} are distances from the model to the scan and from the scan to the model, respectively:

$$E_{m2s}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) = \sum_{i=1}^N \sum_{k=1}^{n_i} \rho(\min_{\mathbf{s} \in \mathcal{S}} \|\hat{\mathbf{p}}_{i,k}(\mathbf{l}_i, \mathbf{r}_i, \mathbf{s}_i, \mathbf{d}_i) - \mathbf{s}\|^2), \quad (7)$$

$$E_{s2m}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) = \sum_{l=1}^S \rho(\min_{\hat{\mathbf{p}}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d})} \|\hat{\mathbf{p}} - \mathbf{s}_l\|^2), \quad (8)$$

where \mathcal{S} is the set of S scan vertices and ρ is the Geman-McClure robust error function [14]. The term $E_{kp}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d})$ is a term for matching model keypoints with scan keypoints, and is defined as the sum of squared distances between corresponding keypoints. This term is important to enabling matching between extremely different animal shapes.

The curvature term E_{curv} favors smoothly connected parts: $E_{curv}(\mathbf{l}, \mathbf{r}, \mathbf{s}, \mathbf{d}) =$

$$k_c \sum_{(i,j) \in \mathcal{C}} \left| \|\hat{\mathbf{n}}_{I_{ij}}(\mathbf{l}_i, \mathbf{r}_i, \mathbf{s}_i, \mathbf{d}_i) - \hat{\mathbf{n}}_{I_{ji}}(\mathbf{l}_j, \mathbf{r}_j, \mathbf{s}_j, \mathbf{d}_j)\|^2 - \|\hat{\mathbf{n}}_{I_{ij}}^{(t)} - \hat{\mathbf{n}}_{I_{ji}}^{(t)}\|^2 \right|, \quad (9)$$

where \mathcal{C} is the set of part connections, $\hat{\mathbf{n}}_{I_{ij}}$ is a vector of vertex normals at the interface points between part i and part j , on part i ; $\hat{\mathbf{n}}_{I_{ji}}$ is a vector of vertex normals at the interface points between part i and part j , on part j . Analogous quantities on the template are denoted with a superscript (t) .

Lastly, E_{pose} is a pose prior on the tail parts learned from animations of the tail defined as the squared Mahalanobis

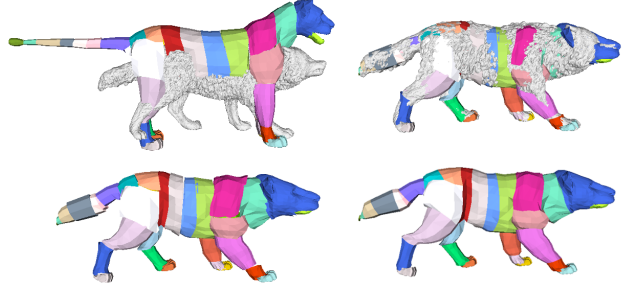


Figure 5: **GLoSS fitting.** Initial template and scan. GLoSS fit to scan. GLoSS model showing the parts. Merged mesh with global topology.

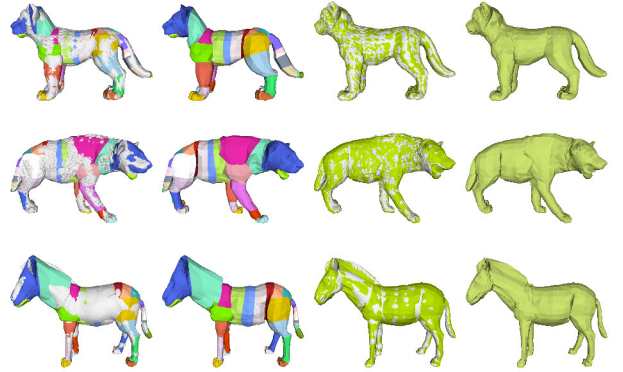


Figure 6: **Registration results.** Comparing GLoSS (left) with the ARAP refinement (right). The fit to the scan is much tighter after refinement.

distance of tail parts pose from a distribution learned from the pose deformation space training set.

We minimize Eq. 4 using the Chumpy auto-differentiation package [1]. Doing so aligns the lioness GLoSS model to all the toys. Figure 5 shows an example of fitting of GLoSS (colored) to a scan (white), and Fig. 6 (first and second column) shows some of the obtained alignments.

ARAP-based refinement. The GLoSS model gives a good initial alignment. Given this, we turn each GLoSS mesh from its part-based topology into a global topology where interface points are not duplicated (Fig. 5). We then further align the vertices \mathbf{v} to the scans by minimizing an energy function defined by a data term similar to Eq. 6 and an As-Rigid-As-Possible (ARAP) regularization [27] term:

$$E(\mathbf{v}) = E_{data}(\mathbf{v}) + E_{arap}(\mathbf{v}). \quad (10)$$

This model-free optimization brings the mesh vertices closer to the scan and therefore captures more accurately the shape of the animal (see Fig. 6).

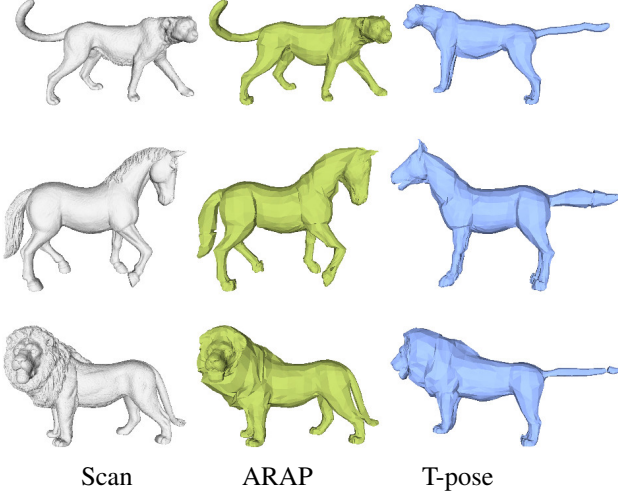


Figure 7: Registrations of toy scans in T-pose.

6. Skinned Multi-Animal Linear Model

The above alignments are now sufficiently accurate to create a first shape model, which we refine further below to produce the full SMAL model.

Pose normalization. GLoSS defines a rotation for each part. Using this, we deform the aligned meshes using linear blend skinning (LBS) to bring them all into the same canonical T-pose. The resulting meshes are not symmetric. This can be due to various reasons: inaccurate pose estimation, limitations of linear-blend-skinning, the toys may not be symmetric, and pose differences across sides of the body create different deformations. We do not want to learn this asymmetry. To address this we perform an averaging of the vertices after we have mirrored the mesh (Fig. 7). Also, the fact that mouths are sometimes open and other times closed presents a challenge for alignment. Finally we smooth the meshes with Laplacian smoothing.

Shape model. Pose normalization removes the non-linear effects of part rotations on the vertices. In the canonical T-pose we can then analyze the statistics of the shape variation in a Euclidean space. We compute the mean shape and the principal components, which capture shape differences between the animals. **SMAL.** The SMAL model is a function $M(\vec{\beta}, \vec{\theta}, \vec{\gamma})$ of shape $\vec{\beta}$, pose $\vec{\theta}$ and translation $\vec{\gamma}$. $\vec{\beta}$ is a vector of the coefficients of the learned PCA shape space, $\vec{\theta} \in \mathbb{R}^{3N} = \{\mathbf{r}_i\}_{i=1}^N$ is the relative rotation of the $N = 33$ joints in the kinematic tree, and $\vec{\gamma}$ is the global translation applied to the root joint. Analogous to SMPL, the SMAL function returns a 3D mesh, where the template model is shaped by $\vec{\beta}$, articulated by $\vec{\theta}$ through LBS, and shifted by $\vec{\gamma}$.

Fitting. To fit SMAL to scans we minimize the following objective:

$$E(\beta, \theta) = E_{pose}(\theta) + E_s(\beta) + E_{data}(\beta, \theta), \quad (11)$$

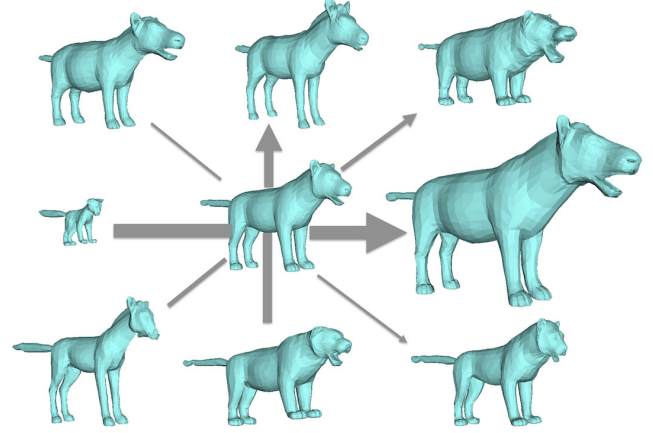


Figure 8: **PCA space.** First 4 principal components. Mean shape is in the center. The width of the arrow represents the order of the components. We visualise deviations of $\pm 2\text{std}$.

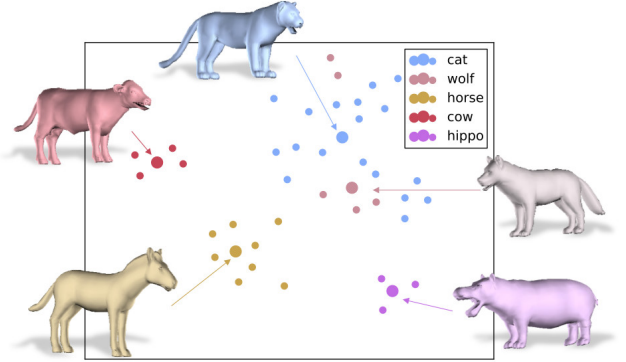


Figure 9: Visualization of different animal families with 8 PCs obtained with t-SNE [21]. Large dots indicate the mean of the PCA coefficients for each family.

where $E_{pose}(\theta)$ and $E_s(\beta)$ are squared Mahalanobis distances from prior distributions for pose and shape, respectively. $E_{data}(\beta, \theta)$ is defined as in Eq. 6 but over the SMAL model. For optimization we use Chumpy [1].

Co-registration. Following [16], we improve the model and registrations by iteratively fitting SMAL to the toys, using this to recompute the registrations, and updating the PCA shape space. The key idea is to *couple*, or regularize, the model-free registration to the model fit by adding this coupling term to Eq. 10:

$$E_{coup}(\mathbf{v}) = k_o |\mathbf{v}_0 - \mathbf{v}|, \quad (12)$$

where \mathbf{v}_0 is the value of the vertices at the end of the SMAL optimization (Eq. 11). During coupling, we use a shape space with 30 dimensions. We perform 4 iterations of registration and model building and observe the registration errors decrease and converge. With the alignments to the toys in the last iteration we learn the shape space of our final SMAL model.

Animal Shape Space. After refining with co-

registration, the final principal components are visualized in Fig. 8. The global shape space in the SMAL model captures the variability of the shapes of animals across different families. The first component captures scale differences; our training set includes adult animals and babies. The learned space nicely separates shape characteristics of animal families. This is illustrated in Fig. 9 with a t-SNE visualization [21] of the first 8 dimensions of the PCA coefficients in the training set. The meshes correspond to the mean shape for each family. We also define family-specific shape models by computing a Gaussian over the PCA coefficients of the class. We compare generic and family specific models below.

7. Fitting Animals to Images

We now fit the SMAL model, $M(\vec{\beta}, \vec{\theta}, \vec{\gamma})$, to image cues by optimizing the shape and pose parameters. We fit the model to a combination of 2D keypoints and 2D silhouettes, both manually extracted, as in previous work [9, 17].

We denote $\Pi(\cdot; f)$ as the perspective camera projection with focal length f , where $\Pi(v_i; f)$ is the projection of the i 'th vertex onto the image plane and $\Pi(M; f) = \hat{S}$ is the projected model silhouette. We assume an identity camera placed at the origin where the global rotation of the 3D mesh is defined by the rotation on the root joint.

In order to fit this model to an image, we formulate an objective function and minimize it with respect to $\Theta = \{\vec{\beta}, \vec{\theta}, \vec{\gamma}, f\}$. Our objective function is a sum of of keypoint and silhouette reprojection error, shape prior and two pose priors, $E(\Theta) =$

$$E_{kp}(\Theta; \vec{x}) + E_{silh}(\Theta; S) + E_{\vec{\beta}}(\vec{\beta}) + E_{\vec{\theta}}(\vec{\theta}) + E_{lim}(\vec{\theta}). \quad (13)$$

Each energy term is weighted by a hyper-parameter defining their importance.

Keypoint reprojection. The keypoint definition of we employ include both points on the surface and inside the surface (e.g. a joint) allowing the keypoint to be labeled in multiple viewpoints. So we assign a set of up to four vertices for each keypoint and take the average of their projection to match the target 2D keypoint. This also makes the method robust to noise in the exact location of the keypoints, since even for humans labeling a point on animals in different poses can be challenging and ambiguous. Specifically for the k 'th keypoint, let \vec{x} be the labeled 2D keypoint and $\{v_{k_j}\}_{j=1}^{k_m}$ be the assigned set of vertices, then

$$E_{kp}(\Theta) = \sum_k \rho\left(\left\|\vec{x} - \frac{1}{|k_m|} \sum_{j=1}^{|k_m|} \Pi(v_{k_j}; \Theta)\right\|_2\right), \quad (14)$$

where ρ is the Geman-McClure robust error function [14].

Silhouette reprojection. We encourage silhouette coverage and consistency similar to [18] using a bi-directional

distance:

$$E_{silh}(\Theta) = \sum_{\vec{x} \in \hat{S}} \mathcal{D}_S(\vec{x}) + \sum_{\vec{x} \in S} \rho(\min_{\vec{\tilde{x}} \in \hat{S}} \|\vec{x} - \vec{\tilde{x}}\|_2). \quad (15)$$

where \mathcal{D}_S is the L2 distance transform field of the data silhouette such that if point \vec{x} is inside the silhouette $\mathcal{D}_S = 0$. Since the silhouette terms have small basins of attraction we optimize the term over multiple scales in a coarse-to-fine manner.

Shape prior: $E_{\vec{\beta}}$. The global shape prior encourages shape coefficients, $\vec{\beta}$, that are probable given the training data. This is the squared Mahalanobis distance with zero mean and variance given by the PCA eigen values. When the animal family is known, we can make our fits more specific by using the mean and the covariance of the training samples of the particular family.

Pose priors. $E_{\vec{\theta}}$ is also defined as the squared Mahalanobis distance using the mean and covariance of the poses across all the training samples and the synthetic walking sequence. To make the pose prior symmetric, we double the training data by reflecting the poses along its main axis. Since we do not have many examples, we further constrain the pose with limit bounds:

$$E_{lim}(\vec{\theta}) = \max(\vec{\theta} - \vec{\theta}_{\max}, 0) + \max(\vec{\theta}_{\min} - \vec{\theta}, 0). \quad (16)$$

$\vec{\theta}_{\max}$ and $\vec{\theta}_{\min}$ are the maximum and minimum range of values for each dimension of $\vec{\theta}$ respectively, which we define by hand. We do not limit the global rotation.

Optimization. Following [6], we first initialize the depth of $\vec{\gamma}$ using the torso points. Then we solve for the global rotation $\{\theta_i\}_{i=0}^3$ and $\vec{\gamma}$ using E_{kp} over points on the torso. Using these as the initialization, we solve Eq. 13 for the entire Θ without E_{silh} . Similar to previous methods [6, 17] we employ a staged approach where the weights on pose and shape prior are gradually lowered over three stages. This helps avoid getting trapped in local optima. We then finally include the E_{silh} term and solve Eq. 13 starting from this initialization. Solving for the focal length is important and we regularize f by adding another term that forces $\vec{\gamma}$ to be close to its initial estimate. The entire optimization is done using OpenDR and Chumpy [19, 1]. Optimization for a single image takes a less than a minute depending on the image on a common Linux machine.

8. Experiments

We have shown how to learn a SMAL animal model from a small set of toy figurines. Now the question is: does this model capture the shape variation of real animals? Here we test this by fitting the model to annotated images of real animals. We fit using class specific, and generic shape models, and show that the shape space generalizes to new animal families not present in training (within reason).



Figure 10: Fits to real images using manually obtained 2D points and segmentation. Colors indicate animal family. We show the input image, fit overlaid, views from -45° and 45° . All results except for those in mint colors use the animal specific shape prior. The SMAL model, learned from toy figurines, generalizes to real animal shapes.

Data. For fitting, we use 19 semantic keypoints of [12] plus an extra point for the tail tip. Note that these keypoints differ from those used in the 3D alignment. We fit frames in the TigDog dataset, reusing their annotation, frames from the Muybridge footage, and images downloaded from the Internet. For images without annotation, we click the same 20 keypoints for all animals, which takes about one minute for each image. We also hand segmented all the images. No images were re-visited to improve their annotations and we found the model to be robust to noise in the exact location of the keypoints. We will make all annotations and results

available.

Results. The model fits to real images of animals are shown in Fig. 1 and 10. The weights for each term in Eq. 13 are tuned by hand and held fixed for fitting *all* images. All results use the animal specific shape space except for those in mint green, which use the global shape model. Despite being trained on scans of toys, our model generalizes to images of real animals, capturing their shape well. Variability in animal families with extreme shape characteristics (*e.g.* lion manes, skinny horse legs, hippo faces) are modeled well. Both the generic and class-specific models capture

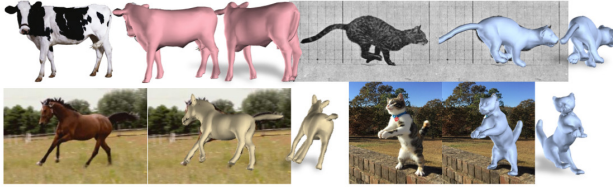


Figure 11: Failure examples due to depth ambiguity in pose and global rotation.

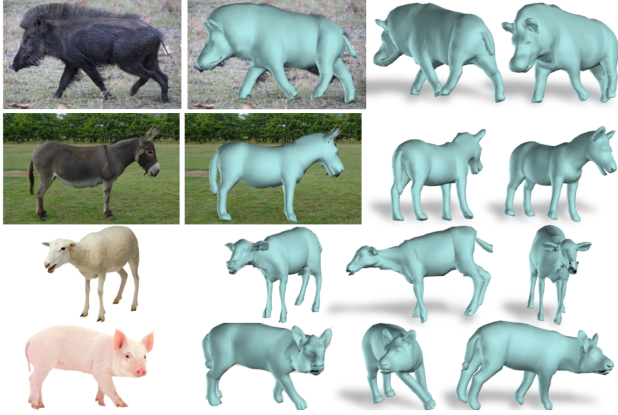


Figure 12: Generalization of SMAL to animal types not present in the shape training set.

the shape of real animals well.

Similar to the case of humans [6], our main failures are due to inherent depth ambiguity, both in global rotation and pose (Fig 11). In Fig. 12 we show the results of fitting the generic shape model to classes of animals not seen in the training set: boar, donkey, sheep and pigs. While characteristic shape properties such as the pig snout cannot be exactly captured, these fits suggest that the learned PCA space can generalize to new animals within a range of quadrupeds.

9. Conclusions

Human shape modeling has a long history, while animal modeling is in its infancy. We have made small steps towards making the building of animal models practical. We showed that starting with toys, we can learn a model that generalizes to images of real animals as well as to types animals that we have not seen before. This gives a procedure for building richer models from more animals and more scans. We hypothesize that the more animals we use to build SMAL, the easier it will be to fit new animals, in a virtuous cycle. Here we fit the model to manually extracted image features but clearly recent work on human 2D pose estimation and segmentation can be extended to animals and this should lead to an automated fitting method.

While we have shown that toys are a good starting point, we would clearly like a much richer model. For that we believe that we need to incorporate image and video evidence.

Our fits to images provide a starting point from which to learn richer deformations to explain 2D image evidence.

Here we have focused on a limited set of quadrupeds. Even among quadrupeds there are many that we do not model. A key issue is dealing with varying numbers of parts (e.g. horns, tusks, trunks) and parts of widely different shape (e.g. elephant ears). Moving beyond the class of animals here, will involve creating a vocabulary of reusable shape parts and new ways of composing them.

Acknowledgments. We thank Seyhan Sitti for scanning the toys.

References

- [1] <http://chumpy.org>.
- [2] Digital life. <http://www.digitallife3d.com/>, Accessed November 12, 2016).
- [3] B. Allen, B. Curless, Z. Popović, and A. Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA '06*, pages 147–156, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.
- [4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: Shape Completion and Animation of PEople. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 24(3):408–416, 2005.
- [5] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *SIGGRAPH*, pages 187–194. ACM, 1999.
- [6] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016, Lecture Notes in Computer Science*. Springer International Publishing, Oct. 2016.
- [7] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, pages 2:690–696, 2000.
- [8] A. Bronstein, M. Bronstein, and R. Kimmel. *Numerical Geometry of Non-Rigid Shapes*. Springer Publishing Company, 2008.
- [9] T. J. Cashman and A. W. Fitzgibbon. What shape are dolphins? building 3d morphable models from 2d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):232–244, Jan 2013.
- [10] Y. Chen, T. Kim, and R. Cipolla. Inferring 3D shapes and deformations from single views. In *iProc. European Conf. on Computer Vision, Part III*, page 300313, 2010.
- [11] Y. Chen, Z. Liu, and Z. Zhang. Tensor-based human body modeling. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 105–112, June 2013.
- [12] L. Del Pero, S. Ricco, R. Sukthankar, and V. Ferrari. Behavior discovery and alignment of articulated object classes from unstructured video. *arXiv preprint arXiv:1511.09319*, 2015.

- [13] L. Favreau, L. Reveret, C. Depraz, and M.-P. Cani. Animal gaits from video. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 277–286. Eurographics Association, 2004.
- [14] S. Geman and D. McClure. Statistical methods for tomographic image reconstruction. *Bulletin of the International Statistical Institute*, 52(4):5–21, 1987.
- [15] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H. Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009.
- [16] D. Hirshberg, M. Loper, E. Rachlin, and M. Black. Coregistration: Simultaneous alignment and modeling of articulated 3D shape. In *European Conf. on Computer Vision (ECCV)*, LNCS 7577, Part IV, pages 242–255. Springer-Verlag, Oct. 2012.
- [17] A. Kanazawa, S. Kovalsky, R. Basri, and D. W. Jacobs. Learning 3D deformation of animals from 2D images. In *Eurographics*, 2016.
- [18] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Category-specific object reconstruction from a single image. In *CVPR*, 2015.
- [19] M. Loper and M. J. Black. OpenDR: An approximate differentiable renderer. In *European Conf. on Computer Vision (ECCV)*, pages 154–169, 2014.
- [20] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015.
- [21] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- [22] D. Marr and K. Nishihara. Representation and recognition of the spatial organization of three dimensional shapes. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 200(1140):269–294, 1978.
- [23] V. Ntouskos, M. Sanzari, B. Cafaro, F. Nardi, F. Natola, F. Pirri, and M. Ruiz. Component-wise modeling of articulated objects. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [24] D. Ramanan, D. A. Forsyth, and K. Barnard. Building models of animals from video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(8):1319–1334, 2006.
- [25] B. Reinert, T. Ritschel, and H.-P. Seidel. Animated 3d creatures from single-view video by skeletal sketching. In *GI '16: Proceedings of the 42st Graphics Interface Conference*, 2016.
- [26] K. Robinette, S. Blackwell, H. Daanen, M. Boehmer, S. Fleming, T. Brill, D. Hoeflerlin, and D. Burnsides. Civilian American and European Surface Anthropometry Resource (CAESAR) final report. Technical Report AFRL-HE-WP-TR-2002-0169, US Air Force Research Laboratory, 2002.
- [27] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing, Barcelona, Spain, July 4-6, 2007*, pages 109–116, 2007.
- [28] D. W. Thompson. *On Growth and Form*. Cambridge University Press, 1917.
- [29] S. Vicente and L. Agapito. Balloon shapes: Reconstructing and deforming objects with volume from images. In *Conference on 3D Vision-3DV*, 2013.
- [30] S. Zuffi and M. J. Black. The stitched puppet: A graphical model of 3D human shape and pose. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2015)*, pages 3537–3546, June 2015.